Справка AppleScript

Основные сведения

О программе AppleScript

Некоторые задачи на Mac часто приходится выполнять многократно. Программа AppleScript позволяет создавать скрипты и программы для автоматизации повторяющихся задач.

AppleScript — это мощный универсальный скриптовый язык, встроенный в OS X. С помощью AppleScript можно создавать сочетания клавиш, автоматизировать повторяющиеся задачи или даже создавать собственные программы, которые помогают сэкономить много времени. AppleScript позволяет управлять скриптуемыми программами.

Скрипт AppleScript состоит из команд, на основе английского языка, которые служат для выполнения различных действий с объектами. Примерами таких действий могут служить: «open», «close» или «get the name of». Примерами объектов являются программы, окна или настройки.

Скрипт может контролировать одну или несколько программ. Для составления скриптов для программы, необходимо, чтобы эта программа распознавала условия, которые используются в скриптах. OS X содержит несколько программ, для которых можно составить скрипты.

Скрипты могут также контролировать программы через сеть, что позволит Вам автоматизировать задачи на одном или нескольких компьютерах.

Вероятно, Вы уже используете AppleScript, даже не догадываясь об этом. Множество функций в системе и программах являются скриптами AppleScript, вызываемыми из какой-либо программы. Например, в этой справочной системе при каждом щелчке по ссылке, предназначенной для открытия программы или файла, для выполнения этого действия запускается скрипт AppleScript.

БЛИЗКИЕ ТЕМЫ

О работе AppleScript
О Редакторе AppleScript
Язык AppleScript
Ресурсы AppleScript в Интернете
Среда разработки AppleScript Studio

Основные сведения

О Редакторе AppleScript

Редактор AppleScript представляет собой удобную в использовании программу работы со скриптами, в которой предусмотрены функции чтения, создания, записи и сохранения скриптов AppleScript.

Ниже перечислены некоторые характеристики Редактора AppleScript.

- Обеспечивает цельный, легкий в использовании интерфейс
- Позволяет находить и заменять текст в скриптах

- Выполняет автоматическую разбивку строк текста длинных скриптовых записей
- Позволяет одновременно отменить несколько действий
- Предоставляет скрипты сочетаний клавиш, обеспечивающие более быстрое написание скрипта
- Поддерживает прикладные службы
- Обеспечивает доступ к скриптам программ из палитры библиотеки
- Включает в себя Ассистент скрипта, упрощающий составление операторов
- Поддерживает функции истории результатов и журнала событий, которые позволяют отслеживать ситуации, связанные со скриптом

БЛИЗКИЕ ТЕМЫ

Язык AppleScript
Среда разработки AppleScript Studio

Основные сведения

Создание скриптов AppleScript

Для работы с AppleScript Вам не требуются знания в области программирования. Вам не придется писать ни одного слова скрипта. В OS X есть сотни полезных скриптов AppleScript, которые Вы можете сразу начать использовать. Кроме того, Вы найдете тысячи скриптов в Интернете.

Вы также можете, взяв за основу имеющиеся скрипты, приспособить их под свои требования посредством минимальных изменений.

Разумеется, наилучшего результата Вы добьетесь, создав собственные скрипты. Научившись писать скрипты, Вы сможете создавать эффективные инструменты и автоматизированные процессы, которые сделают работу более продуктивной.

Редактор AppleScript позволяет изменять скрипты и создавать собственные скрипты в простой и удобной среде. Все примеры в Справке AppleScript созданы с помощью Редактора AppleScript.

- Откройте Редактор AppleScript.

  Введите следующий текст в окне Редактора AppleScript.

  ```
  say "Welcome to AppleScript!"
  ```

  Нажмите кнопку запуска.

БЛИЗКИЕ ТЕМЫ

О Редакторе AppleScript
Среда разработки AppleScript Studio

Основные сведения

О работе AppleScript

Работа AppleScript осуществляется путем отправки сообщений, называемых «событиями Apple». При запуске скрипта AppleScript отправляет события Apple в систему или программам, выполняемым команды.

В большинстве случаев программы или система отправит результат в AppleScript. Например, если скрипт отправляет запрос программе для получения имени объекта, программа выдает имя. Если команда не может быть выполнена, программа или AppleScript выдают сообщение об ошибке. Возвращенные результаты отображаются в Редакторе AppleScript.

Как правило, знать принцип работы событий Apple необязательно. AppleScript выполняет все действия в фоновом режиме. Вам необходимо только составлять и запускать скрипты.

БЛИЗКИЕ ТЕМЫ

О программе AppleScript
Язык AppleScript

Основные сведения

Об апплетах

Скрипт, сохраненный в качестве программы, называется программой «AppleScript» или апплетом. Апплеты автоматически запускаются при открытии и выполняют команды скрипта. Апплеты могут представлять собой небольшие функции, содержащие всего несколько команд, или сложными программами с тысячами строк.

Апплеты можно сохранить в виде редактируемых или только выполняемых программ. Редактируемый апплет содержит текст скрипта, что позволяет в любое время выполнить его редактирование и рекомпиляцию. Для редактирования или изменения апплета перетащите его на значок Редактора AppleScript. При сохранении скрипта в виде только выполняемого, редактируемая часть удаляется, после чего файл уже не может быть отредактирован или рекомпилирован.

Для редактирования апплета, который был сохранен с возможностью редактирования, следует открыть его в редакторе AppleScript или в другой программе редактирования AppleScript. Апплетам необходим обработчик «on run», начинающий выполнение скрипта при открытии апплета.

- В этом примере показан обработчик «on run».

```
on run
        --first command
        --second command
end run
```

БЛИЗКИЕ ТЕМЫ

О дроплетах
Типы файлов AppleScript

Основные сведения

О компилированных скриптах

Скомпилированный скрипт представляет собой файл AppleScript в редактируемой форме, который можно

просматривать и править в редакторе AppleScript или в другой программе редактирования AppleScript.

При открытии скомпилированного скрипта он отображается в Редакторе AppleScript, но его команды не выполняются автоматически. Это позволит осуществить любые изменения скрипта без необходимости его фактического выполнения. Затем Вы можете компилировать скрипт, читающий и преобразовывающий весь текст в код AppleScript внутри файла. Скомпилированный скрипт можно выполнить. Компилированный скрипт имеет следующее расширение файла: .scpt.

При сохранении файла в качестве компилированного скрипта текст, наряду с кодом, по-прежнему остается в файле. Поэтому в любой момент можно снова отредактировать скомпилированный скрипт.

БЛИЗКИЕ ТЕМЫ

Об апплетах
О дроплетах
Типы файлов AppleScript

О дроплетах

Дроплет представляет собой скрипт AppleScript, который выполняется при перетягивании файлов или папок на его значок. Скрипт выполняет действия с этими файлами или папками.

Так же как и апплеты, дроплет можно сохранить в редактируемой или только выполняемой форме. Редактируемый дроплет содержит текст скрипта, что позволяет в любое время выполнить его редактирование и рекомпиляцию. Чтобы редактировать или изменить дроплет, перетяните его на значок Редактора скрипта. При сохранении скрипта в виде только выполняемого, редактируемая часть удаляется, после чего файл уже не может быть отредактирован или рекомпилирован.

Дроплетам необходим обработчик «on open», начинающий выполнение скрипта при открытии дроплета. Обработчик «on open» имеет параметр, содержащий список объектов, перетянутых в дроплет. Часть скрипта дроплета служит для анализа списка, а другая часть — для выполнения действий с объектами. За исключением случаев, когда дроплет разработан для выполнения действия только над одним объектом, он должен содержать циклический элемент, обеспечивающий выполнение действий над каждым объектом списка.

- Вот пример обработчика дроплета «on open» и списком параметров.

```
on open (theItemsDropped)
        --first command
        --second command
end open
```

БЛИЗКИЕ ТЕМЫ

Об апплетах
О компилированных скриптах
Типы файлов AppleScript

Работа со словарями AppleScript

Чтобы выяснить, какие команды и объекты Вы можете использовать для управления программой, обратитесь к словарю этой программы. Словарь содержит набор терминов, которые понятны для программы.

Для просмотра словаря программы в Редакторе AppleScript можно выбрать программу в окне «Библиотека» или перетащить значок программы на значок Редактора AppleScript.

Термины в словаре объединены в группы родственных терминов, называемые «пакетами». Команды и объекты выделены жирным шрифтом.

---

1. В Редакторе AppleScript выберите пункт меню «Окно» > «Библиотека».
2. Дважды нажмите имя программы в окне «Библиотеки».
3. Выберите пакет.

   Отобразятся термины этого пакета и их описания.

В языке AppleScript предусмотрены следующие категории.

- Пакеты: Группы родственных терминов. В словаре редактора AppleScript пакеты отображаются в первом столбце и обозначаются буквой S оранжевого цвета.
- Команды: запрашивают действие или результат. Команды, называемые также «событиями», отображаются во втором столбце и обозначаются буквой C синего цвета (которая относится также к событиям).
- Классы: Объекты, для которых можно написать скрипт. В описании класса перечислены свойства и элементы, существующие в данном классе. Классы отображаются во втором столбце и обозначаются буквой C сиреневого цвета.
- Свойства: Объекты, содержащие значения. Свойства отображаются в третьем столбце и обозначаются буквой P сиреневого цвета.
- Элементы: Объекты, содержащиеся в других объектах. Например, параграф может содержать множество слов. Элементы могут отображаться в третьем столбце и обозначаются буквой сиреневого цвета E оранжевого цвета.

БЛИЗКИЕ ТЕМЫ

Определения классов
Определения команд
Управление программами с помощью AppleScript

ОТКРЫТЬ

AppleScript Language Guide: Commands Reference
AppleScript Language Guide: Class Reference

Устранение неполадок

Если у Вас возникли проблемы с AppleScript, попробуйте найти информацию об устранении неполадок в

следующих разделах справки.

Если при нажатии кнопки «Отменить» в диалоговом окне «Выбрать из списка» отображается сообщение об ошибке

Если действие папки запускается самопроизвольно

Если скрипт не распознает том

Определения классов

Определение класса описывает свойства и элементы объекта, управляемого скриптом. Класс можно сравнить с существительным в предложении, а определение класса – с определением существительного в словаре.

Как правило, определение класса в словаре AppleScript начинается с термина, выделенного полужирным шрифтом; после термина в той же строке приведено его краткое описание.

Далее следует список элементов или объектов, которые принадлежат объекту, поддерживающему скрипты, или составляют его. Список элементов не всегда включен во все определения класса, т.к. некоторые объекты, управляемые скриптом, содержат другие управляемые скриптом объекты, а другие – нет.

Определение класса заканчивается списком всех свойств и качеств объекта, управляемого скриптом, а также из значений.

Свойство начинается с имени, за которым следует:

- Описание формата данных, в котором хранится значение свойства, например, целое (число), значение с точкой (список из двух чисел), строка (текстовые данные) или логическое значение (true или false)
- Необязательный индикатор типа [r/o], указывающий на то, что свойство имеет значение «только для чтения» (его можно прочитать, но невозможно изменить)
- Краткое описание свойства

Если объект имеет множественное число, Вы можете ссылаться сразу на несколько объектов.

Если объект содержит список элементов, то он содержит другие объекты. Имя элемента, которое Вы можете использовать в скрипте, выделено жирным шрифтом. Обычный текст описывает, каким образом можно сослаться на элемент.

Объекты могут обладать свойствами. Имена свойств выделены жирным шрифтом. Обычный текст описывает тип значения, содержащегося в свойстве, например, строку. Свойства могут представлять собой постоянное значение или один из списков констант. Скрипт может получить значение свойства или ввести значение свойства, если свойство содержит «[r/o]» (read-only) после синтаксической структуры.

БЛИЗКИЕ ТЕМЫ

Определения команд
Определения объектов
Команды AppleScript
Работа со словарями AppleScript

ОТКРЫТЬ

AppleScript Language Guide: Class Reference

Определения команд

Команды — это термины AppleScript, выполняющие действия. Команды в словаре AppleScript отображаются в центральном столбце и обозначаются буквой C синего цвета. Определение команды включает краткое описание действия, выполняемого командой, а также ее синтаксическую структуру. Синтаксическая структура – это способ написания команды в скрипте. Синтаксическая структура может состоять из нескольких частей.

Слова, выделенные жирным шрифтом, – это термины, используемые в операторе. Записывать термины нужно в точности так, как они записаны в словаре.

Слова, выделенные синим цветом, передают предоставленную Вами информацию. Например, Вы можете узнать указанный Вами объект или переменную.

Некоторые команды содержат необязательные фрагменты, отображаемые в квадратных скобках. Использовать их в операторе необязательно.

БЛИЗКИЕ ТЕМЫ

Определения классов
Работа со словарями AppleScript
Определения событий
Определения объектов

ОТКРЫТЬ

AppleScript Language Guide: Commands Reference

Определения событий

Определение события описывает глагол или команду, используемые для задания какого-либо действия в объекте, управляемом скриптом. События представляют собой действия или глаголы, которые могут быть выполнены объектами или в которых они могут участвовать.

Как правило, определение события начинается с термина, за которым в той же строке следует краткое описание термина.

За описанием следует список всех параметров, используемых при выполнении команды. Факультативные параметры заключены в скобки; обязательные параметры – нет.

Каждый параметр начинается с имени, за которым следует формат данных, используемый с параметром, и заканчивается коротким описанием параметра.

Толкование команды содержит краткое описание (подчеркнутый текст) действия команды, а также ее синтаксическую структуру. Синтаксическая структура – это способ написания команды в скрипте. Синтаксическая структура может состоять из нескольких частей.

Слова, выделенные жирным шрифтом, – это термины, используемые в операторе. Записывать термины нужно в точности так, как они записаны в словаре.

Слова обычного текста передают предоставленную Вами информацию. Например, Вы можете узнать указанный

Вами объект или переменную.

Некоторые команды содержат необязательные фрагменты, отображаемые в квадратных скобках. Использовать их в операторе необязательно.

Команды применяются к объектам. Толкование объекта отображает синтаксис, по которому можно узнать объект, включая его имя и краткое описание (подчеркнутый текст). Класс объекта отображает определенные характеристики объекта, например, команды, которые могут быть применимы к объекту.

БЛИЗКИЕ ТЕМЫ

События базы данных
Определения классов
Определения команд
Определения объектов
О работе AppleScript

Определения объектов

Команды применяются к объектам. Определение объекта в словаре AppleScript содержит синтаксис для распознавания объекта, включая его имя и краткое описание (подчеркнутый текст). Класс объекта отображает определенные характеристики объекта, например, команды, которые могут быть применимы к объекту.

Если объект имеет множественное число, Вы можете ссылаться сразу на несколько объектов.

Если объект содержит список элементов, то он содержит другие объекты. Имя элемента, которое Вы можете использовать в скрипте, выделено жирным шрифтом. Обычный текст описывает, каким образом можно сослаться на элемент.

Объекты могут обладать свойствами. Имена свойств выделены синим жирным шрифтом. Обычный текст синего цвета описывает тип значения, содержащегося в свойстве, например, строку. Свойства могут представлять собой постоянное значение или один из списков констант. Скрипт может получить значение свойства или установить значение свойства, если свойство содержит «[r/o]» (read-only) после синтаксической структуры.

БЛИЗКИЕ ТЕМЫ

Управление программами с помощью AppleScript
Определения команд
Определения классов
Задание действий объекту
Работа со словарями AppleScript

ОТКРЫТЬ

AppleScript Language Guide: Class Reference
AppleScript Language Guide: Свойства

Операторы if

Оператор if — это команда, которая выполняется только при соблюдении определенных условий. Каждый

оператор if содержит логическое выражение, которое может иметь значение true (истина) или false (ложь). Команды, содержащиеся в операторе if, выполняются только в том случае, если логическое выражение имеет значение true. Условный оператор оканчивается словом «if». Например:

```
if amountReceived > 100 then
        set memberStatus to "special"
end if
```

Команды в приведенном выше операторе if выполняются, если логическое выражение имеет значение true (полученная сумма больше 100). Если логическое выражение имеет значение false, команды не выполняются.

Оператор if может также содержать оператор else. Такая комбинация известна как оператор if-then-else. Например:

```
if amountReceived > 100 then
        set memberStatus to "special"
else
        set memberStatus to "general"
end if
```

Если первое логическое выражение в приведенном выше операторе имеет значение true (полученная сумма больше 100), выполняются команды в первой части оператора. Если первое логическое выражение имеет значение false, выполняются команды во второй части оператора.

Вы можете поместить операторы if один в другой, чтобы задействовать несколько условий.

Подробную информацию об операторах if в AppleScript Вы найдете в AppleScript Language Guide.

БЛИЗКИЕ ТЕМЫ

Операторы repeat
Определения команд
Ввод переменных в AppleScript
Постоянные значения

ОТКРЫТЬ

AppleScript Language Guide – If

Операторы repeat

Операторы repeat позволяют выполнять задачи несколько раз, в зависимости от условий. Оператор repeat обеспечивает циклическое выполнение группы операторов, продолжающееся до тех пор, пока не будет выполнено условие завершения цикла. Вы можете поместить операторы if один в другой, чтобы отвечать на изменения условий.

В AppleScript доступен широкий выбор операторов repeat, отвечающих на различные условия.

---

repeat (forever)

Повторять циклически. Завершить такой цикл можно с помощью оператора exit.

```
repeat
	-- statement
	-- statement
	if (x = 5) then exit repeat
end repeat
```

---

repeat (число) times

Определенное число повторов.

```
repeat 5 times
	-- statement
	-- statement
end repeat
```

---

repeat while

Повторять, пока верно определенное условие.

```
set x to 1
repeat while (x < 5)
	-- statement
	-- statement
	set x to (x + 1)
end repeat
```

---

repeat until

Повторять, пока не станет верным определенное условие.

```
set x to 1
repeat until (x = 5)
	-- statement
	-- statement
	set x to (x + 1)
end repeat
```

---

repeat with (переменная) from (начальное значение) to (конечное значение)

Переменной присваивается начальное значение, а после каждого цикла это значение увеличивается на единицу. Повтор заканчивается, когда переменная достигает конечного значения.

```
repeat with x from (5) to (25)
	-- statement
	-- statement
end repeat
```

---

repeat with (переменная) in (список)

Устанавливает переменную на место первого объекта в списке. Каждый раз при прохождении цикла переменная

устанавливается на место следующего объекта. Повтор завершается, когда использованы все объекты списка.

```
set myList to {234, 342, 681, 874}
repeat with x in myList
        -- statement
        -- statement
end repeat
```

Дополнительную информацию об операторах repeat в AppleScript смотрите в руководстве «AppleScript Language Guide».

БЛИЗКИЕ ТЕМЫ

Операторы if
Ввод переменных в AppleScript
Постоянные значения

ОТКРЫТЬ

AppleScript Language Guide - repeat

Основные команды

Постоянные значения

Константа – это зарезервированное слово в AppleScript, которое содержит заданное значение. AppleScript содержит множество констант, которые можно использовать для получения информации об объекте, проведению сравнений или выполнению арифметических вычислений. Программы и другие объекты могут задать собственные константы. Эти константы задаются в определениях класса значений.

Чтобы увидеть значение константы, вставьте этот пример в Редактор AppleScripts и затем нажмите «Запустить». Чтобы увидеть значение константы, нажмите «Результат».

Результат тоже является константой. Эта константа относится к скрипту и указывает значение, возвращаемое командой.

```
get the current date
```

Ниже приведены другие примеры констант:

- Арифметические константы: число пи
- БулеВы константы: верно (true), неверно (false)
- Константы даты и времени: минуты, часы, дни, недели
- Строковые константы: return, space, tab
- Скриптовые константы: anything, current application, it, me, missing value, my, result

Дополнительную информацию о константах AppleScript смотрите в руководстве «AppleScript Language Guide».

БЛИЗКИЕ ТЕМЫ

Ввод переменных в AppleScript
Определения объектов

Определения классов
Определения событий

ОТКРЫТЬ

AppleScript Language Guide - constants

Основные команды

Ввод переменных в AppleScript

Переменная – это container-объект, созданный в скрипте, который используется для хранения значения. При ее создании Вы задаете имя переменной, которое также называется идентификатором. Во время запуска скрипта Вы можете присвоить значение переменной, получить доступ к значению или изменить его. Переменные сохраняются только на время выполнения скрипта. Когда скрипт завершен, значение утрачивается.

- В этом примере показано, как указывается уникальный идентификатор и ему назначается значение.

```
set myVar to "153"

set x to "Quarterly Reports"

set x to false
```

Чтобы увидеть, как значения хранятся в переменной, вставьте эти примеры в Редактор AppleScripts и затем нажмите «Запустить». Для просмотра значения переменной выберите «Результат».

Дополнительную информацию о переменных в AppleScript смотрите в руководстве «AppleScript Language Guide».

БЛИЗКИЕ ТЕМЫ

Постоянные значения
Операторы if
Операторы repeat

ОТКРЫТЬ

AppleScript Language Guide - variables

Основные команды

Указание размещения программы

Программы в Mac OS X структурированы в виде «пакетов». Пакет — это специальная папка, которая содержит папки и файлы, составляющие данную программу. Поэтому для программ Mac OS X команды path to и info for, могут возвращать различные результаты.

При применении команды path to к программе путь, как правило, оканчивается расширением .app. Имена файлов могут отображаться в Finder без расширений, однако и в этом случае возможно распознавание файла в скрипты по его расширению.

При получении информации о программе с помощью команды info for свойство Folder для результата, как

правило, имеет значение true. Дополнительные свойства результата, такие как extension hidden, name extension и displayed name, содержат дополнительную информацию о файле и его имени.

БЛИЗКИЕ ТЕМЫ

path to
info for
Ссылки на файлы, папки и диски
choose application
Скриптовые команды Finder

Ссылки на файлы, папки и диски

AppleScript поддерживает ссылки на некоторые папки, файлы и диски по имени без указания пути. Местоположение таких объектов, как папка «Документы», рабочий стол, папка пользователя и загрузочный диск, может быть определено средствами AppleScript.

- В этом примере показана ссылка на определенную папку с использованием ее имени в скрипте.

```
set x to the path to the desktop folder
```

БЛИЗКИЕ ТЕМЫ

Ссылка на Finder в скриптах
path to
info for
Скриптовые команды Finder

Ссылка на Finder в скриптах

Для работы с командами Finder следует направить команды в Finder с помощью оператора tell так же, как они направляются программе.

- Вот пример использования оператора tell в скрипте. Оператор tell направляет функцию number of windows в программу Finder.app.

```
tell application "Finder.app"
        return the number of windows
end tell
```

БЛИЗКИЕ ТЕМЫ

Finder
Задание действий объекту
Ссылки на файлы, папки и диски
Определения объектов

Скриптовые команды Finder

ОТКРЫТЬ

AppleScript Language Guide - Object specifiers

Команды Finder

Скриптовые команды Finder

Вот некоторые команды, которые можно использовать с Finder:

activate

alias file

alias list

application file

clean up

close

column

Computer - object

container

copy

count

data size

delete

desktop window

desktop

disk

document file

duplicate

eject

empty trash

exists

file

file information

find

Finder

Finder window

folder

get

icon view options

information window

Internet location file

item

list view options

make

move

moving folder window for

open

package

path to

preferences

preferences window

print

quit

Ссылки на файлы, папки и диски

Ссылка на Finder в скриптах

reveal

select

set

sort

trash

update

window

Стандартные команды скриптовых дополнений

Скриптовые дополнения добавляют многие полезные команды, которые можно включать в скрипты.

adding folder items to

beep

choose application

choose color

choose file

choose file name

choose folder

choose from list

choose URL

clipboard info

close access

closing folder window for

current date

delay

dialog reply

display

display dialog

do shell script

download

FTP item

get eof

info for

list disks

list folder

load script

mount volume

offset

open for access

open location

opening folder

path to

POSIX path

random number

read

round

run script

say

scripting components

set eof

set the clipboard to

set volume

store script

summarize

system attribute

time to GMT

upload

URL

write